UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/993,865	11/14/2001	William M. Cullen	23982-10313	8307

89188 7590 12/22/2009

Progress / Fenwick
Silicon Valley Center
801 California Street
Mountain View, CA 94041

EXAMINER	
PATEL, DHAIRYA A	

ART UNIT	PAPER NUMBER
2451	

NOTIFICATION DATE	DELIVERY MODE
12/22/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ptoc@fenwick.com
gsueoka@fenwick.com
nmorad@fenwick.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/993,865
Filing Date: November 14, 2001
Appellant(s): CULLEN ET AL.

Greg T. Sueoka (Reg. # 33,800)
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 9/3/2009 appealing from the Office action mailed 2/3/2009.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed subsequent to final rejection mailed on February 3, 2009.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,397,352 Chandrasekaran et al.

5,596,720 Hamada et al.

5,850,525 Kalkunte et al.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1,3,7-12,16-17,20-25,28-31 rejected under 35 U.S.C. 103(a) as being unpatentable over Chandrasekaran et al. U.S. Patent # 6,397,352 (hereinafter Chandrasekaran) in view of Hamada et al. U.S. Patent # 5,596,720 (hereinafter Hamada) further in view of Kalkunte et al. U.S. Patent # 5,850,525 (hereinafter Kalkunte)

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1,3,7-12,16-17,20-25,28-31 rejected under 35 U.S.C. 103(a) as being unpatentable over Chandrasekaran et al. U.S. Patent # 6,397,352 (hereinafter Chandrasekaran) in view of Hamada et al. U.S. Patent # 5,596,720 (hereinafter Hamada) further in view of Kalkunte et al. U.S. Patent # 5,850,525 (hereinafter Kalkunte)

As per claim 1, Chandrasekaran teaches a method of handling a message received at a messaging system server, the method comprising:

-storing, in non-persistent storage, the message; (Fig. 2A element 204) (column 6 lines 61-67) (column 7 lines 1-2). **NOTE:** The reference teaches the message is stored in the propagation queue (non-persistent storage).

-determining whether the message has been delivered (column 11 lines 8-22);

NOTE: The reference teaches storing in a table within non-volatile memory a message w/ UID value and initial states "prepared". The prepared state indicates the message has been sent but that an acknowledge message has not yet been received for the message from the destination site (determining whether the message has been delivered) since the acknowledgement has not been received, it means it is determining if the message has been delivered.

-if the message has been delivered, removing the message from the non-persistent storage (column 12 lines 24-31) **NOTE:** The reference teaches after the receiving message A in the destination site, message A will be dequeued from the received message queue (i.e. non-persistent storage).

- after a delay interval has elapsed (column 12 lines 12-13), and if the message has not been removed from the non-persistent storage, saving the message to persistent storage (column 7 lines 31-39)(column 10 lines 43-67)(column 11 lines 1-22)(column 12 lines 4-11, lines 33-42).

NOTE: In column 10 lines 43-67, column 11 lines 1-22, Chandrasekaran teaches the message is in the propagation queue which is non-persistent storage (volatile) and transmitting the message to the destination site, and still storing the message in the propagation queue (Fig. 3 element 312)(storing message in non-

persistent storage) because the acknowledge message has not yet been received (responsive to the attempt not being successful), the propagation process then receives message data (content of the message) to store in durable or non-volatile memory (persistent storage) at the source site and by maintaining the propagated message data in a nonvolatile memory (column 7 lines 30-39) a recovery mechanism is provided that allows the source site to determine whether the message has been sent to the destination site.

Chandrasekaran teaches after the delay interval has elapsed saving the message to the persistent storage (column 7 lines 31-39)(column 10 lines 43-67)(column 11 lines 1-22) but is silent in teaching so that message can be retrieved and delivered. Hamada teaches saving the message to the persistent storage so that message can be retrieved and delivered (Fig. 21 element 101-5, 201-5)(column 17 lines 35-65)(Fig. 23). It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Hamada's teaching in Chandrasekaran's teaching to come up storing message in non-persistent storage for an interval and retrieving and delivering the message that is save in the persistent storage. The motivation for doing so would be so the message can be retrieved from the non-volatile memory and retransmitted or re-sent to the receiver or the destination at a later time, therefore non-volatile/persistent storage is used to save the message for later retransmission.

Chandrasekaran, and Hamada teaches if the message has not been removed from the non-persistent storage, saving the message to persistent

storage so that message can be retrieved and delivered but are silent in teaching after a configurable delay interval has elapsed.

Kalkunte teaches after a configurable delay interval (column 5 lines 15-44)(column 6 lines 65-67). It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Kalkunte's teaching in Chandrasekaran and Hamada's teaching to come up with having a delay interval that is configurable. The motivation for doing so would be so that the user can set the amount to wait for the delay interval, before saving the message to the persistent storage.

As per claim 3, Chandrasekaran, Hamada and Kalkunte teaches the method of claim 1, but Chandrasekaran further teaches wherein storing in the non-persistent storage comprises storing in a log queue. (Fig. 2A element 204) (Column 6 lines 61-67) (Column 7 lines 1-2).

As per claim 7, Chandrasekaran, Hamada and Kalkunte teaches the method of claim 1, but Kalkunte further teaches further comprising determining the delay interval. (column 6 lines 65-67). **NOTE:** The reference teaches calculating the delay time (determining the delay interval).

As per claim 8, Chandrasekaran, Hamada and Kalkunte teaches the method of claim 7, but Kalkunte further teaches wherein determining the delay interval comprises: determining at least one metric based on messages handled by the server (column 5 lines 28-44); and determining the delay interval based on the at least one metric (column 6 lines 65-67) (column 5 lines 28-44) **NOTE:** The reference metric which in this case is delay time and/or value tow.

As per claim 9, Chandrasekaran, Hamada and Kalkunte teaches the method of claim 8, but Kalkunte further teaches wherein the metric comprises a metric based on a number of sending clients using the server to deliver messages (column 5 lines 28-44)

As per claim 10, Chandrasekaran, Hamada and Kalkunte teaches the method of claim 7, but Kalkunte further teaches wherein determining the delay interval comprises dynamically determining the delay (column 6 lines 65-67)

As per claim 11, Chandrasekaran, Hamada teaches the method of claim 1, but Chandrasekaran further teaches wherein the message was received over a communications network. (Fig. 13 element 728,722,726,720)

As per claim 12, Chandrasekaran, Hamada teaches the method of claim 1, but Chandrasekaran further teaches wherein the messages comprise a guaranteed messages; (column 7 lines 39-57) and wherein the messaging system comprises a message-oriented middleware system. (Column 7 lines 27-38) **NOTE:** The reference teaches sends the commit messages (guaranteed messages) to the destination site to indicate the transaction (transferring) should be committed. The reference also teaches that messages are maintained in the non-volatile memory at the source site until they are transferred to the destination site. Therefore in case of the source site failure, destination site will fetch the message from the non-volatile memory from the source site.

As per claim 16, Chandrasekaran teaches a computer program product, disposed on a computer readable medium, for handling messages received at a

server, the computer program including instructions for causing a server processor to:

-store, in a non-persistent storage (Fig. 2A element 204), messages received from at least one client as the messages are received; (column 6 lines 61-67) (column 7 lines 1-2)(column 12 lines 4-11) **NOTE:** The reference teaches the message is stored in the propagation queue (non-persistent storage), and it also teaches upon receiving the message storing the message in received message queue (column 12 lines 4-11).

-determining whether one of the guaranteed messages has been delivered (column 11 lines 8-22); **NOTE:** The reference teaches storing in a table within non-volatile memory a message w/ UID value and initial states “prepared”. The prepared state indicates the message has been sent but that an acknowledge message has not yet been received for the message from the destination site (determining whether the message has been delivered) since the acknowledgement has not been received, it means it is determining if the message has been delivered.

-if the guaranteed message has been delivered, removing the message from the non-persistent storage The reference teaches after the receiving message A in the destination site, message A will be dequeued from the received message queue (i.e. non-persistent storage).

-after a delay interval has elapsed and if the message has not removed from the non-persistent storage, save the message to persistent storage (column

7 lines 31-39) (column 10 lines 43-67) (column 11 lines 1-22) (column 12 lines 4-11, lines 33-42).

NOTE: In column 10 lines 43-67, column 11 lines 1-22, Chandrasekaran teaches the message is in the propagation queue which is non-persistent storage (volatile) and transmitting the message to the destination site, and still storing the message in the propagation queue (Fig. 3 element 312)(storing message in non-persistent storage) because the acknowledge message has not yet been received (responsive to the attempt not being successful), the propagation process then receives message data (content of the message) to store in durable or non-volatile memory (persistent storage) at the source site and by maintaining the propagated message data in a nonvolatile memory (column 7 lines 30-39) a recovery mechanism is provided that allows the source site to determine whether the message has been sent to the destination site.

Chandrasekaran teaches after the delay interval has elapsed saving the message to the persistent storage (column 7 lines 31-39)(column 10 lines 43-67)(column 11 lines 1-22) but is silent in teaching so that message can be retrieved and delivered. Hamada teaches saving the message to the persistent storage so that message can be retrieved and delivered (Fig. 21 element 101-5, 201-5)(column 17 lines 35-65)(Fig. 23). It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Hamada's teaching in Chandrasekaran's teaching to come up storing message in non-persistent storage for an interval and retrieving and delivering the message that is save in the persistent storage. The motivation for doing so would be so

the message can be retrieved from the non-volatile memory and retransmitted or re-sent to the receiver or the destination at a later time, therefore non-volatile/persistent storage is used to save the message for later retransmission.

Chandrasekaran, and Hamada teaches if the message has not been removed from the non-persistent storage, saving the message to persistent storage so that message can be retrieved and delivered but are silent in teaching after a configurable delay interval has elapsed.

Kalkunte teaches after a configurable delay interval (column 5 lines 15-44)(column 6 lines 65-67). It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Kalkunte's teaching in Chandrasekaran and Hamada's teaching to come up with having a delay interval that is configurable. The motivation for doing so would be so that the user can set the amount to wait for the delay interval, before saving the message to the persistent storage.

As per claim 17, Chandrasekaran, Hamada and Kalkunte teaches a computer program of claim 16, but Chandrasekaran further teaches wherein the instructions for causing the server processor to store messages in the non-persistent storage comprise instructions for causing the server processor to store the messages in a log queue (Fig. 2A element 204)(column 6 lines 61-67)(column 7 lines 1-2).

As per claim 20, Chandrasekaran, Hamada and Kalkunte teaches a computer program of claim 16, but Kalkunte further teaches further comprising instructions for causing the server processor to determine the delay. (column 6

lines 65-67). **NOTE:** The reference teaches calculating the delay time (determining the delay interval).

As per claim 21, Chandrasekaran teaches the computer program of claim 20, wherein the instructions for causing the server processor to determine the delay comprise instructions for causing the server processor to: determine at least one metric based on the received messages (column 5 lines 28-44); and determine the delay based on the at least one metric (column 6 lines 65-67)(column 5 lines 28-44) **NOTE:** The reference metric which in this case is delay time and/or value tow.

As per claim 22, Chandrasekaran teaches the computer program of claim 21, wherein the metric comprises a metric based on a number of clients using the server to deliver messages (column 5 lines 28-44)

As per claim 23, Chandrasekaran and Hamada teaches a computer program of claim 16, but Chandrasekaran further teaches wherein the instructions for causing the processor to determine the delay comprise instructions for causing the processor to dynamically determining the delay. (Column 8 lines 20-47)

As per claim 24-25,28-31, they teach same limitations as claims 1-17,20-23 respectively, therefore rejected under same basis.

1. Claims 4-6,13-15,18-19,26-27 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Chandrasekaran in view of Hamada further in view of Kalkunte further in view of Stein et al. U.S. Patent 6,289,212 (hereinafter Stein).

As per claim 4, Chandrasekaran, Hamada and Kalkunte teaches the method of claim 1, but fails to teach further comprising transmitting an acknowledgement message to a client that sent the received message, the acknowledgement message indicating that the received message will not be lost by the server in the case of server failure. Stein teaches transmitting an acknowledgement message to a client that sent a received message, the acknowledgement message indicating that the received message will not be lost by the server in the case of server failure. (Column 12 lines 21-37). It would have obvious to one of ordinary skill in the art at the time of applicant's invention to implement Stein's teaching in Chandrasekaran, Hamada and Kalkunte's teaching to come up with transmitting an acknowledgement message. The motivation for transmitting the acknowledgement message is to let the user know that the message has been sent and be delivered.

As per claim 5, Chandrasekaran, Hamada and Kalkunte fails to teach the method of claim 4, wherein transmitting the acknowledgment message to the client comprises transmitting the acknowledgment message to the client for successful delivery of the received message. Stein teaches transmitting the acknowledgment message to the client comprises transmitting the acknowledgment message to the client for successful delivery of the received message (Column 12 lines 21-37) It would have obvious to one of ordinary skill in the art at the time of applicant's invention to implement Stein's teaching in Chandrasekaran, Hamada and Kalkunte's teaching to come up with transmitting an acknowledgement message. The motivation for transmitting the

acknowledgement message is to let the user know that the message has been sent and be delivered.

As per claim 6, Chandrasekaran, Hamada and Kalkunte fails to teach the method of claim 4, wherein transmitting the acknowledgment message to the client comprises transmitting the acknowledgment message to the client for storage of the received message in persistent storage. Stein teaches the method of claim 4, wherein transmitting an acknowledgment message to the client comprises transmitting the acknowledgment message to the client for the storage of the received message in persistent storage. (Column 12 lines 21-37). The reference teaches that message is sent and the facsimile message is placed in the asynchronous request queue. It would have obvious to one of ordinary skill in the art at the time of applicant's invention to implement Stein's teaching in Chandrasekaran, Hamada and Kalkunte's teaching to come up with transmitting the acknowledgement message for the received message's storage in persistent storage. The motivation for doing so would have been so that the client knows that message is going to be delivered properly.

As per claim 13, Chandrasekaran teaches a method of handling guaranteed messages received at a message-oriented middleware server over a network, the method comprising: storing, in a log queue in non-persistent storage (Fig. 2 element 204) guaranteed messages received from at least one client as the guaranteed messages are received (column 6 lines 61-67) (column 7 lines 1-2)(column 12 lines 4-11) **NOTE:** The reference teaches the message is stored in the propagation queue (non-persistent storage), and it also teaches upon

receiving the message storing the message in received message queue (column 12 lines 4-11).

-determining whether one of the guaranteed messages has been delivered (column 11 lines 8-22);

NOTE: The reference teaches storing in a table within non-volatile memory a message w/ UID value and initial states "prepared". The prepared state indicates the message has been sent but that an acknowledge message has not yet been received for the message from the destination site (determining whether the message has been delivered) since the acknowledgement has not been received, it means it is determining if the message has been delivered.

-if the guaranteed message has been delivered, removing the message from the non-persistent storage (column 12 lines 24-31) **NOTE:** The reference teaches after the receiving message A in the destination site, message A will be dequeued from the received message queue (i.e. non-persistent storage).

-after delay period has elapsed and saving the guaranteed message to persistent storage (column 7 lines 31-39) (column 10 lines 43-67) (column 11 lines 1-22).

In column 10 lines 43-67, column 11 lines 1-22, Chandrasekaran teaches the message is in the propagation queue (log queue) which is non-persistent storage (volatile) and transmitting the message to the destination site, and still storing the message in the propagation queue (Fig. 3 element 312)(storing message in non-persistent storage) because the acknowledge message has not yet been received (responsive to the attempt not being successful) and after a

delay interval the propagation process then receives message data (content of the message) to store in durable or non-volatile memory (persistent storage) at the source site and by maintaining the propagated message data in a nonvolatile memory (column 7 lines 30-39)(emphasis added) a recovery mechanism is provided that allows the source site to determine whether the message has been sent to the destination site.

Chandrasekaran fails to teach if attempting to deliver one of the guaranteed message was not successful, continuing to store the guaranteed message in the non-persistent storage and after the determined delay period has elapsed saving the guaranteed message to the persistent storage so that the guaranteed message can be retrieved and delivered. Hamada teaches if attempting to deliver one of the guaranteed message was not successful, continuing to store the guaranteed message in the non-persistent storage (column 7 lines 55-64) and after the determined delay period has elapsed saving the guaranteed message to the persistent storage so that the guaranteed message can be retrieved and delivered (Fig. 21 element 101-5, 201-5)(column 17 lines 35-65)(Fig. 23). It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Hamada's teaching in Chandrasekaran's teaching to come up storing message in non-persistent storage for an interval and retrieving and delivering the message that is save in the persistent storage. The motivation for doing so would be so the message can be retrieved from the non-volatile memory and retransmitted or re-

sent to the receiver or the destination at a later time, therefore non-volatile/persistent storage is used to save the message for later retransmission.

Chandrasekaran, Hamada teaches a delay time period, but are silent in teaching dynamically determining a delay time period and the determined delay time period. Kalkunte teaches dynamically determining a delay time period and the determined delay time period (column 5 lines 15-44)(column 6 lines 65-67). It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Kalkunte's teaching in Chandrasekaran and Hamada's teaching to come up with having a delay interval that is configurable. The motivation for doing so would be so that the user can set the amount to wait for the delay interval, before saving the message to the persistent storage.

Chandrasekaran, Hamada and Kalkunte fails to teach transmitting a guarantee acknowledgement message to a client that sent the received guaranteed message whose delivery was attempted, the guarantee acknowledgement message indicating that the message will not be lost by the server. Stein teaches transmitting a guarantee acknowledgement message to a client that sent the received guaranteed message whose delivery was attempted, the guarantee acknowledgement message indicating that the message will not be lost by the server. (column 12 lines 21-37). The reference teaches that the facsimile message has been sent is a guaranteed message indicating the message is not going to be lost because if the other side would not receive the fax, the message has been sent would not be displayed. It would have obvious to

one of ordinary skill in the art at the time of applicant's invention to implement Stein's teaching in Chandrasekaran, Hamada and Kalkunte's teaching to come up with transmitting an acknowledgement message. The motivation for transmitting the acknowledgement message is to let the user know that the message has been sent and be delivered.

As per claim 14, Chandrasekaran, Hamada, Kalkunte and Stein teaches the method of claim 13, but Chandrasekaran, Hamada and Kalkunte fails to teach transmitting the guarantee acknowledgement message comprises: if the guaranteed message was successfully delivered, transmitting the guarantee acknowledgement message; and if the guaranteed message was not successfully delivered, transmitting the guarantee acknowledgement message when the guaranteed message is persistently stored. Stein teaches transmitting the guarantee acknowledgement message comprises if the guaranteed message was successfully delivered, transmitting the guarantee acknowledgement message (Column 12 lines 21-37); and if the guaranteed message was not successfully delivered, transmitting the guarantee acknowledgement message when the guaranteed message is persistently stored (Column 12 lines 21-37). It would have obvious to one of ordinary skill in the art at the time of applicant's invention to implement Stein's teaching in Chandrasekaran, Hamada and Kalkunte's teaching to come up with acknowledgement message when the message is persistently stored if the guaranteed message not successfully delivered. The motivation for doing so would have to let the user know that the message has been received by source site and will be delivered properly.

As per claim 15, Chandrasekaran, Hamada, Kalkunte and Stein teaches the method of claim 13, but Kalkunte further teaches wherein dynamically determining the delay time period comprises: determining a metric based on messages handled by the server (column 5 lines 28-44); and determining the delay time period based on the determined metric (column 6 lines 65-67) (column 5 lines 28-44) **NOTE:** The reference metric which in this case is delay time and/or value tow.

As per claim 18, Chandrasekaran, Hamada and Kalkunte teaches the computer program of claim 16, but fails to teach further comprising instructions for causing the server processor to transmit an acknowledgement message to a client that sent the received message whose delivery was attempted, the acknowledgement message indicating that the received message will not be lost by the server. Stein teaches instructions for causing the server processor to transmit an acknowledgement message to a client that sent a received message whose delivery was attempted, the acknowledgement message indicating that the received message will not be lost by the server. (Column 12 lines 21-37). It would have obvious to one of ordinary skill in the art at the time of applicant's invention to implement Stein's teaching in Chandrasekaran, Hamada and Kalkunte's teaching to come up with transmitting an acknowledgement message. The motivation for transmitting the acknowledgement message is to let the user know that the message has been sent and be delivered.

As per claim 19, Chandrasekaran and Hamada fails to teach the computer program of claim 18, wherein the instructions for causing the server processor to

transmit the acknowledgment message to the client comprise instructions for causing the server processor to transmit the acknowledgment message to the client for a message saved from non-persistent storage to persistent storage. Stein teaches the computer program of claim 18, wherein the computer program instructions for causing the server processor to transmit an acknowledgment message to the client comprise instructions for causing the server processor to transmit the acknowledgment message to the client for a message saved from non-persistent storage to persistent storage. (Column 12 lines 21-37). The reference teaches that message is sent and the facsimile message is placed in the asynchronous request queue. It would have obvious to one of ordinary skill in the art at the time of applicant's invention to Stein's teaching in Chandrasekaran, Hamada and Kalkunte's teaching to come up with transmitting the acknowledgement message for messages saved to persistent storage. The motivation for doing so would have been so that the client knows that message is going to be delivered properly.

As per claim 26-27 they teach same limitations as claim 18,19 respectively, therefore rejected under same basis.

(10) Response to Argument

Appellant's Argument:

Appellant states Chandrasekaran does not disclose, teach or suggest "if the message has been delivered, removing the message from the non-persistent storage" because determining a site failure does not disclose confirming the message's delivery.

Examiner's Response:

As per appellant's argument, Examiner respectfully disagrees with the appellant because in Fig. 2b, Fig. 3 and column 10 lines 58-67 and column 11 lines 1-2 , Chandrasekaran teaches the message, the assigned propagation sequence number and the information are sent to the destination site (sending the message) (column 10 lines 58-64). After the transmission of the message (i.e. message has been delivered), the propagated message information is stored in the non-volatile memory (column 10 lines 65-67) (column 11 lines 1-2). Chandraseksaran also states that this is a separate transaction to durably store the fact that the message is sent to the destination site.

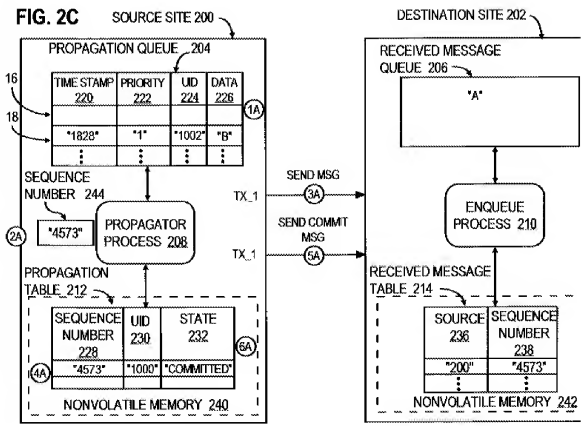
At step **312**, the message, the assigned propagation sequence number and information that identifies source site as the propagating site, are sent to the destination site. For example, as indicated by state "3A", as part of TX_1, the message for entry **216** and the propagation sequence number value of "4573" is transmitted from source site **200** to destination site **202**.

At block **320** a second transaction ("TX_2") is initiated at the source site for storing the propagated message information in nonvolatile memory. The second transaction is

US 6,3

11

performed as a separate transaction to durably store the fact that the message has been sent to the destination site. Because TX_2 is performed as a separate transaction, once



Furthermore, in column 12 lines 24-31, Chandrasekaran teaches after the receiving message A at the destination site (this means that the message has been delivered) which is sent from the source site, message A will be dequeued from the received message queue (i.e. non-persistent storage). Chandrasekaran specifically states receiving message “A”, and then dequeing message “A” from message queue at the source site. The message queue is non-persistent storage. Appellant states requirement for removing the message in Chandrasekaran is completely different from the present invention. Examiner respectfully disagrees and would like to point out that claim language states "if

the message has been delivered", Chandrasekaran in column 12 lines 24-25 states receiving message "A" at the destination site" and then claim language states "removing the message from the non-persistent storage", Chandrasekaran teaches dequeuing the message after it has been delivered from the received message queue at the source site which means the message is removed from the non-persistent storage after it has been delivered. The removing of the message step is conducting after the message has been received/delivered in Chandrasekaran.

> For example, if destination site **202** determines after receiving message "A" that source site **200** has failed, destination site **202** rolls back TX_3 thus causing message "A" to be dequeued from received message queue **206**. By dequeuing message "A", destination site **202** may continue to execute as if it never received message "A" from source site **200**, as it is guaranteed that message "A" will be retransmitted upon the recovery of source site **200**.
>
> At step **410**, upon receiving a commit message from the source site, the received message data is stored into non-volatile memory at destination site **102**. For example, upon receiving a commit message from source site **200** (state 5A), as illustrated by state "2B" in FIG. 2D, a source identifier information "200" and the propagation sequence number "4573" are stored in received message table **214** in non-volatile memory **242**. This information can be used in the event of a failure to determine whether a particular message was previously received at destination site **202** from source site **200**.

Furthermore, on page 6 of the appeal brief filed on 9/3/2009, Appellant also admits that message is transmitted to destination side (7:28-30) and then the message information i.e. propagation sequence number, UID, are then stored in the non-volatile memory in the propagation table (moving the message to persistent storage after the message is delivered) from the propagation queue (i.e. removing from the non-persistent storage). This step of storing the message

data into the non-volatile memory is done after the message has been transmitted to a destination site. Therefore, this means that after the message is delivered, the removing the message from non-persistent storage. Therefore Chandrasekaran teaches the claimed limitations.

Appellant's Argument:

Appellant states Chandrasekaran does not disclose, teach or suggest "after a configurable delay interval has elapsed and if the message has not been removed from the non-persistent storage, saving the message to persistent storage so that message can be retrieved and delivered".

Examiner's Response:

Examiner respectfully disagrees with the appellant because in column 10 lines 43-67, column 11 lines 1-22, column 12 lines 4-11, lines 33-42, Chandrasekaran teaches after a delay period (column 12 lines 12-13), if the message has not been removed from the non-persistent storage, saving the message to persistent storage. Chandrasekaran teaches the message is in the propagation queue which is non-persistent storage (volatile) and transmitting the message to the destination site, and still storing the message in the propagation queue (Fig. 3 element 312)(storing message in non-persistent storage) because the acknowledge message has not yet been received (responsive to the attempt not being successful), the propagation process then receives message data (content of the message) to store in durable or non-volatile memory (persistent storage) at the source site and by maintaining the propagated message data in a nonvolatile memory (column 7 lines 30-39) a recovery mechanism is provided

that allows the source site to determine whether the message has been sent to the destination site. After enqueing the message (this means message has been retrieved), the destination site returns and an acknowledgement to the source site to indicate the enqueue operation has been performed (column 7 lines 40-50). This means the saving of the data/message in the non-volatile memory (persistent storage) is so the message can be retrieved and delivered. Examiner would also like to point out that Chandrasekaran particularly states that destination site ***waits*** for a commits message and after that the message(column 12 lines 12-13), so that the received message is stored in the non-volatile memory (column 12 lines 32-40). This is considered a delay period has elapsed i.e. waiting for the message.

At step **408**, the destination site waits for a commit message to be received from the source site. However, if the

At step **410**, upon receiving a commit message from the source site, the received message data is stored into non-volatile memory at destination site **102**. For example, upon receiving a commit message from source site **200** (state 5A), as illustrated by state "2B" in FIG. **2D**, a source identifier information "200" and the propagation sequence number "4573" are stored in received message table **214** in non-volatile memory **242**. This information can be used in the event of a failure to determine whether a particular message was previously received at destination site **202** from source site **200**.

According the Fig. 2D, it shows message A is still not removed from the received message queue (non-persistent storage) (step 1B) and after a wait for commit message (steps 5A), then the message is stored at non-volatile memory (persistent storage) (step 2b) and (Fig. 4 step 410).

Applicant states Chandrasekaran teaches saving propagated message data to the persistent storage. Examiner would like to point out, Although, Chandrasekaran teaches saving propagated message data in the persistent storage, it is used to retrieved and delivered at a later time incase of source site failure (column 7 lines 30-50). Therefore, Chandrasekaran does teach storing message in non-volatile memory so the message can be retrieved and delivered.

> nation site. The propagation process then stores the propagation sequence number, the UID and an initial propagation state ("propagated message data") into a propagation table that is maintained in durable (nonvolatile) memory at the source site. By maintaining the propagated message data in nonvolatile memory, a recovery mechanism is provided that allows the source site to determine, even after a source site failure, whether a particular message has previously been propagated to the destination site.
>
> Upon receiving the message from the source site, the destination site enqueues the message for execution and stores the propagation sequence number and the identity of the source site ("received message data") in a received message table that is maintained in durable (nonvolatile) memory at the destination site. The information maintained in the received message table provides a mechanism that allows the destination site, even after a destination failure, to determine whether a particular message has previously been received from the source site. After enqueuing the received message, the destination site returns an acknowledgment to the source site to indicate that it has performed the enqueue.

Examiner would also like to point out that the claim language states "storing message" in the non-volatile memory which can be any message i.e. it can be broadly interpreted to include propagated message data. Claim language does not specify that the storing the "original message sent". Therefore, Chandrasekaran teaches the claimed limitations.

Appellant's Argument:

Hamada does not disclose and Hamada cannot be impermissibly modified to disclose "if the message has not been removed from the non-persistent storage, saving the message to persistent storage so that the message can be retrieved and delivered".

Examiner's Response:

As per remark C, Examiner respectfully disagrees w/ the applicant because in column 17 lines 35-65, Hamada teaches saving the message to the persistent storage so that message can be retrieved and delivered (Fig. 21 element 101-5, 201-5)(column 17 lines 35-65)(Fig. 23). Hamada specifically states, the message ID and the content of the message are also stored in non-volatile memory. In column 6 lines 35-39, Hamada teaches saving the message to the non-volatile memory, therefore if the failure occurs the information is recalled to the volatile memory. In column 8 lines 9-13, Hamada teaches when the message is received from the client, the server checks whether the process from client has been completed by making a reference to the volatile memory which means the message is stored in the volatile memory of client initially. When it is determined the process is not yet performed, the message is stored in non-volatile memory (column 8 lines 17-23). This means if the message is not removed from the volatile memory (non-persistent storage), then saving the message to the non-volatile memory (i.e. persistent storage). Furthermore, Hamada also teaches in upon server failure, the recalling of the data from the

non-volatile memory to the volatile memory is also done (column 8 lines 25-30).

Therefore Hamada teaches the claimed limitations.

When the message from the client is received, the server **30**, initially checks whether the demanded process from the client **10** has been completed by making reference to the volatile memory **36**, as show in step **ST1** of the flowchart of FIG. **4**. According to checking process at the step **ST1**, if it is determined that the demanded process has already been completed, the process is advanced to step **ST2** to issue a notice notifying that the process has already been completed to the client **10**. On the other hand, when it is determined that the demanded process is not yet processed according to the checking process of step **ST1**, the process is advanced to step **ST3**, in which the demanded process is performed (executed by the server demand processing stage **302**), then, at step **ST4**, the result of the process is written in the non-volatile memory for dealing with the server failure. Subsequently, at step **ST5**, a notice for normal, completion is issued to the client **10**. Once the server failure occurs, upon resumption of processing after the failure, the process is performed to realize continuation of the process of message reception by recalling the management data of the result of the process maintained in the non-volatile memory **39** to the volatile memory **36**.

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Hamada's teaching in Chandrasekaran's teaching to come up storing message in non-persistent storage for an interval and retrieving and delivering the message that is save in the persistent storage. The motivation for doing so would be so the message can be retrieved from the non-volatile memory and retransmitted or re-sent to the receiver or the destination at a later time, therefore non-volatile/persistent storage is used to save the message for later retransmission.

Appellant's Argument:

Kalkunte does not disclose and Kalkunte cannot impermissibly modified to disclose "after a configurable delay interval has elapsed…saving the message to persistent storage so that the message can be retrieved and delivered.

Examiner's Response:

Examiner respectfully disagrees with the applicant because Chandrasekaran and Hamada teaches if the message has not been removed from the non-persistent storage, saving the message to persistent storage so that message can be retrieved and delivered (as stated above in the other Examiner's response), but both are silent in teaching after a configurable delay interval has elapsed i.e. configuring the delay interval. Kalkunte teaches after a configurable delay interval (column 5 lines 15-44) (column 6 lines 65-67). Kalkunte teaches having the propagation delay interval which is can be any value ***upto*** 2.56 microseconds based on the value of t. This means this a configurable delay interval based on the formula listed below of t_d. Therefore Kalkunte teaches configurable delay interval.

accessing the media **50**. Each of the clients is assigned a delay time that includes the IPG interval plus a randomly selected interval within the range of τ and 2τ, whereby the value τ equals the maximum one-way propagation delay between the corresponding client **44** and the server **42**. Thus, the value of τ for each client **44** will vary depending on the topology of the network **40**.

management of the present invention. If desired, the server **42** may include the value τ, up to a maximum of 2.56 microseconds on a 100 megabit per second hub, in the message to the clients **44**. The message to the clients **44** may also include a propagation delay interval (ΔX) to be used for collision mediation. As described below, the propagation delay interval (ΔX) represents the maximum propagation delay between any two stations **44***a* and **44***b*, i.e., the maximum time needed between any two stations to know that there is activity on the network. The value τX can be up to 2.56 microseconds for a 100 megabyte per second network.

$t_d = A\Delta X + Ct_s$.

Thus, the delay time t_d is calculated in step **92** based on a two-dimensional randomized selection of the slot time t_s and

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention was made to implement Kalkunte's teaching in Chandrasekaran and Hamada's teaching to come up with having a delay interval that is configurable to remove message from non-persistent storage and saving to the persistent storage. The motivation for doing so would be so that the user can set the amount to wait (time period) for the delay interval, before saving the message to the persistent storage manually.

Appellant also states that combination of Kalkunte with Chandrasekaran and Hamada is a classic example of impermissible hindsight reconstruction.

In response to appellant's argument that the examiner's conclusion of obviousness is based upon improper hindsight reasoning, it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the

applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Dhairya A Patel/

Examiner, Art Unit 2451

/John Follansbee/

Supervisory Patent Examiner, Art Unit 2451

Conferees:

/Jeffrey Pwu/

Supervisory Patent Examiner, Art Unit 2446

/John Follansbee/

Supervisory Patent Examiner, Art Unit 2451